

(19)



JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

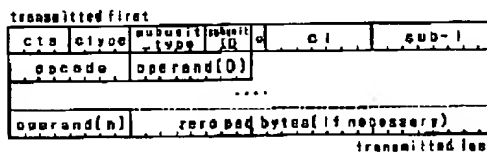
(11) Publication number: **10341247 A**(43) Date of publication of application: **22 . 12 . 98**(51) Int. Cl. **H04L 12/40**(21) Application number: **09152677**(22) Date of filing: **10 . 06 . 97**(71) Applicant: **SONY CORP**(72) Inventor: **KOJIMA TAKASHI
AOKI YUKIHIKO**(54) **DATA TRANSMITTER, DATA RECEIVER, DATA
TRANSMISSION SYSTEM AND DATA
TRANSMISSION METHOD**

(57) Abstract:

PROBLEM TO BE SOLVED: To allow the system to conduct transmission/reception of data whose packet size exceeds one packet efficiently by adopting a format for each packet that is made up of a format dependent storage field, a packet label field storing relating information used for split transmission of data, and a data field storing a command to a transmission destination device and/or optional data.

SOLUTION: Not only a command is sent simply but also transmission data are transmitted while being divided into a plurality of packets by adopting a new command frame that adds a field to an AV/C command frame as one format for an FCP frame specified by a cts field of a command packet. Return data are set to a field equivalent to an operand field of an AV/C response frame in the case that not only a response is singly returned by the new response frame but also optional return data are returned.

COPYRIGHT: (C)1998,JPO



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-341247

(43) 公開日 平成10年(1998)12月22日

(51) Int.Cl.⁶

H 0 4 L 12/40

識別記号

F I

H 0 4 L 11/00

3 2 0

審査請求 未請求 請求項の数29 O L (全 22 頁)

(21) 出願番号 特願平9-152677

(22) 出願日 平成9年(1997)6月10日

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72) 発明者 小嶋 隆嗣

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(72) 発明者 青木 幸彦

東京都品川区北品川6丁目7番35号 ソニー株式会社内

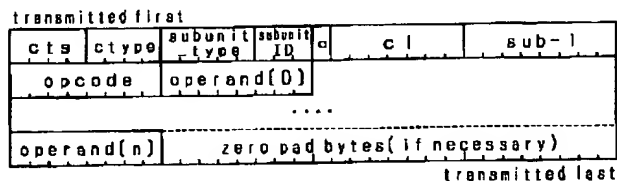
(74) 代理人 弁理士 小池 晃 (外2名)

(54) 【発明の名称】 データ送信装置、データ受信装置、データ伝送システム及びデータ伝送方法

(57) 【要約】

【課題】 パケットによってやり取りする際に、コマンドの送受信だけでなく、1パケットに納まりきらないようなデータの送受信も、コマンドの送受信に使用されるものと同様なパケットによって効率良く行えるようにする。

【解決手段】 パケットに、パケット内のフレームのフォーマットを示す情報が格納されるフィールドを設ける。そして、このフィールドによって識別されるフォーマットのひとつとして、送信データを複数のパケットに分割して送信する際にそれらのパケットの関係を示す情報が格納されるパケットラベルフィールドと、送信先機器に対するコマンド及び／又は任意のデータが格納されるデータフィールドとを上記フレームが有するように規定するフォーマットを設ける。



本発明で使用するコマンドフレームの一例

【特許請求の範囲】

【請求項 1】 所定のフォーマットのフレームを含むパケットによってデータを送信するデータ送信装置であって、

上記フレームは、当該フレームのフォーマットの種別を示す情報が格納されるフィールドを有し、

上記フォーマットの一つとして、送信データを複数のパケットに分割して送信する際にそれらのパケットの関係を示す情報が格納されるパケットラベルフィールドと、送信先機器に対するコマンド及び／又は任意のデータが格納されるデータフィールドとを上記フレームが有するように規定するフォーマットを備えていることを特徴とするデータ送信装置。

【請求項 2】 上記パケットラベルフィールドは、送信データを複数のパケットに分割して送信する際に、それらのパケットに共通の値が設定されるフィールドを有することを特徴とする請求項 1 記載のデータ送信装置。

【請求項 3】 上記パケットラベルフィールドは、送信データを複数のパケットに分割して送信する際に、それらのパケットの順序を示す情報が格納されるフィールドを有することを特徴とする請求項 1 記載のデータ送信装置。

【請求項 4】 上記パケットラベルフィールドは、送信データを複数のパケットに分割して送信する際に、最後のパケットであるか否かを示す情報が格納されるフィールドを有することを特徴とする請求項 1 記載のデータ送信装置。

【請求項 5】 上記パケットは、IEEE 1394 で規定された Asynchronous パケットであることを特徴とする請求項 1 記載のデータ送信装置。

【請求項 6】 上記フレームは、IEC 1883 で規定された FCP フレームであることを特徴とする請求項 1 記載のデータ送信装置。

【請求項 7】 受信したパケットに応じて、所定のフォーマットのフレームを含むパケットによって返信データを返すデータ受信装置であって、

上記フレームは、当該フレームのフォーマットの種別を示す情報が格納されるフィールドを有し、

上記フォーマットの一つとして、返信データを複数のパケットに分割して送信する際にそれらのパケットの関係を示す情報が格納されるパケットラベルフィールドと、受信したパケットに対するレスポンス及び／又は任意のデータが格納されるデータフィールドとを上記フレームが有するように規定するフォーマットを備えていることを特徴とするデータ受信装置。

【請求項 8】 上記パケットラベルフィールドは、返信データを複数のパケットに分割して送信する際に、それらのパケットに共通の値が設定されるフィールドを有することを特徴とする請求項 7 記載のデータ受信装

置。

【請求項 9】 上記パケットラベルフィールドは、返信データを複数のパケットに分割して送信する際に、それらのパケットの順序を示す情報が格納されるフィールドを有することを特徴とする請求項 7 記載のデータ受信装置。

【請求項 10】 上記パケットラベルフィールドは、返信データを複数のパケットに分割して送信する際に、最後のパケットであるか否かを示す情報が格納されるフィールドを有することを特徴とする請求項 7 記載のデータ受信装置。

【請求項 11】 上記パケットは、IEEE 1394 で規定された Asynchronous パケットであることを特徴とする請求項 7 記載のデータ受信装置。

【請求項 12】 上記フレームは、IEC 1883 で規定された FCP フレームであることを特徴とする請求項 7 記載のデータ受信装置。

【請求項 13】 所定のフォーマットのフレームを含む送信用パケットによってデータを送信するデータ送信装置と、

上記送信用パケットを受信したときに、当該送信用パケットによって送信された送信データに応じた返信データを、所定のフォーマットのフレームを含む送信用パケットによって、上記データ送信装置へ送信するデータ受信装置とを有し、

上記送信用パケットに含まれるフレームは、当該フレームのフォーマットの種別を示す情報が格納されるフィールドを有し、

上記送信用パケットに含まれるフレームのフォーマットの一つとして、送信データを複数のパケットに分割して送信する際にそれらのパケットの関係を示す情報が格納されるパケットラベルフィールドと、上記データ受信装置に対するコマンド及び／又は任意のデータが格納されるデータフィールドとを当該フレームが有するように規定するフォーマットを備えていることを特徴とするデータ送受信システム。

【請求項 14】 上記送信用パケットのパケットラベルフィールドは、

送信データを複数の送信用パケットに分割して送信する際に、それらの送信用パケットに共通の値が設定されるフィールドと、

送信データを複数の送信用パケットに分割して送信する際に、それらの送信用パケットの順序を示す情報が格納されるフィールドと、

送信データを複数の送信用パケットに分割して送信する際に、最後の送信用パケットであるか否かを示す情報が格納されるフィールドとを有することを特徴とする請求項 13 記載のデータ送受信システム。

【請求項 15】 上記送信用パケットに含まれるフレームは、当該フレームのフォーマットの種別を示す情報が

格納されるフィールドを有し、
上記返信用パケットに含まれるフレームのフォーマットの一つとして、返信データを複数のパケットに分割して送信する際にそれらのパケットの関係を示す情報が格納されるパケットラベルフィールドと、受信したパケットに対するレスポンス及び／又は任意のデータが格納されるデータフィールドとを当該フレームが有するように規定するフォーマットを備えていることを特徴とする請求項 1 3 記載のデータ送受信システム。

【請求項 1 6】 上記返信用パケットのパケットラベルフィールドは、
返信データを複数の返信用パケットに分割して送信する際に、それらの返信用パケットに共通の値が設定されるフィールドと、
返信データを複数の返信用パケットに分割して送信する際に、それらの返信用パケットの順序を示す情報が格納されるフィールドと、
返信データを複数の返信用パケットに分割して送信する際に、最後の返信用パケットであるか否かを示す情報が格納されるフィールドとを有することを特徴とする請求項 1 5 記載のデータ送受信システム。

【請求項 1 7】 上記送信用パケットは、IEEE 1394 で規定された Asynchronous パケットであることを特徴とする請求項 1 3 記載のデータ送受信システム。

【請求項 1 8】 上記送信用パケットに含まれるフレームは、IEC 1883 で規定された FCP フレームであることを特徴とする請求項 1 3 記載のデータ送受信システム。

【請求項 1 9】 上記返信用パケットは、IEEE 1394 で規定された Asynchronous パケットであることを特徴とする請求項 1 3 記載のデータ送受信システム。

【請求項 2 0】 上記返信用パケットに含まれるフレームは、IEC 1883 で規定された FCP フレームであることを特徴とする請求項 1 3 記載のデータ送受信システム。

【請求項 2 1】 所定のフォーマットのフレームを含む送信用パケットによって送信側ノードから受信側ノードへ送信データを送るとともに、上記送信用パケットを受信側ノードが受け取ったら、所定のフォーマットのフレームを含む返信用パケットによって受信側ノードから送信側ノードへ返信データを送ることにより、ノード間でデータの伝送を行うデータ伝送方法において、
上記送信用パケットに含まれるフレームは、当該フレームのフォーマットの種別を示す情報が格納されるフィールドを有し、
上記送信用パケットに含まれるフレームのフォーマットの一つとして、送信データを複数のパケットに分割して送信する際にそれらのパケットの関係を示す情報が格納されるパケットラベルフィールドと、受信側ノードに対するコマンド及び／又は任意のデータが格納されるデー

タフィールドとを当該フレームが有するように規定するフォーマットを備えていることを特徴とするデータ伝送方法。

【請求項 2 2】 上記送信用パケットのパケットラベルフィールドは、
送信データを複数の送信用パケットに分割して送信する際に、それらの送信用パケットに共通の値が設定されるフィールドと、
送信データを複数の送信用パケットに分割して送信する際に、それらの送信用パケットの順序を示す情報が格納されるフィールドと、
送信データを複数の送信用パケットに分割して送信する際に、最後の送信用パケットであるか否かを示す情報が格納されるフィールドとを有することを特徴とする請求項 2 1 記載のデータ伝送方法。

【請求項 2 3】 上記返信用パケットに含まれるフレームは、当該フレームのフォーマットの種別を示す情報が格納されるフィールドを有し、
上記返信用パケットに含まれるフレームのフォーマットの一つとして、返信データを複数のパケットに分割して送信する際にそれらのパケットの関係を示す情報が格納されるパケットラベルフィールドと、受信したパケットに対するレスポンス及び／又は任意のデータが格納されるデータフィールドとを当該フレームが有するように規定するフォーマットを備えていることを特徴とする請求項 2 1 記載のデータ伝送方法。

【請求項 2 4】 上記返信用パケットのパケットラベルフィールドは、
返信データを複数の返信用パケットに分割して送信する際に、それらの返信用パケットに共通の値が設定されるフィールドと、
返信データを複数の返信用パケットに分割して送信する際に、それらの返信用パケットの順序を示す情報が格納されるフィールドと、
返信データを複数の返信用パケットに分割して送信する際に、最後の返信用パケットであるか否かを示す情報が格納されるフィールドとを有することを特徴とする請求項 2 3 記載のデータ伝送方法。

【請求項 2 5】 上記送信用パケットは、IEEE 1394 で規定された Asynchronous パケットであることを特徴とする請求項 2 1 記載のデータ伝送方法。

【請求項 2 6】 上記送信用パケットに含まれるフレームは、IEC 1883 で規定された FCP フレームであることを特徴とする請求項 2 1 記載のデータ伝送方法。

【請求項 2 7】 上記返信用パケットは、IEEE 1394 で規定された Asynchronous パケットであることを特徴とする請求項 2 1 記載のデータ伝送方法。

【請求項 2 8】 上記返信用パケットに含まれるフレームは、IEC 1883 で規定された FCP フレームであることを特徴とする請求項 2 1 記載のデータ伝送方法。

【請求項 29】 上記送信用パケット及び返信用パケットをノード間でやり取りするに当たって、予めノード間で論理的なコネクションを設定するとともに当該コネクションの中に論理的なチャネルを設定し、送信データを送信用パケットによって受信側ノードへ送る際は、使用するコネクション及びチャネルの識別情報を送信用パケットに付加し、返信データを返信用パケットによって送信側ノードへ送る際は、使用するコネクション及びチャネルの識別情報を返信用パケットに付加することを特徴とする請求項 21 記載のデータ伝送方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、パケットによってデータの送受信を行うデータ送信装置、データ受信装置、データ伝送システム及びデータ伝送方法に関し、特に IEEE 1394 の Asynchronous パケットによるデータの送受信に好適なデータ送信装置、データ受信装置、データ伝送システム及びデータ伝送方法に関する。

【0002】

【従来の技術】 パーソナルコンピュータや、ハードディスク装置等の外部記憶装置や、プリンタ等の出力装置や、デジタルビデオディスクプレーヤ又はデジタルスチルカメラ等の AV 機器などを相互に接続してデータの送受信を行うバスとして、Institute of Electrical and Electronics Engineers 1394（以下、IEEE 1394 と称する。）によって規定された IEEE 1394 High Performance SerialBus（以下、IEEE 1394 バスと称する。）がある。IEEE 1394 バスでは、接続されている各ノードに対して ID 番号を割り付け、それらのノード間において、パケットによってデータの送受信を行う。

【0003】 IEEE 1394 バスのモードには、周期的にパケットの送受信を行う Isochronous モードと、周期に関係なくパケットの送受信を行う Asynchronous モードとがある。Isochronous モードは、125 μ s 毎のデータ転送を保証しており、画像データや音声データのよう、一定時間毎に一定量のデータ転送を行う必要があるようなデータの転送に適している。一方、Asynchronous モードは、例えば、IEEE 1394 バスによって接続された機器間でコマンドをやり取りする際に使用される。

【0004】 従来、Asynchronous モードでは、1 コマンドパケットに対して 1 レスポンスパケットが対応するようになされている。すなわち、ノード A からノード B にコマンドを送信する際は、図 19 に示すように、先ず、ノード A からノード B に、ノード B に対するコマンドを含むコマンドパケットを送信する。そして、コマンドパケットを受信したノード B は、コマンドパケットに対するレスポンスとして、ノード A へレスポンスパケットを

送信する。

【0005】 具体的には、例えば、ノード A からノード B へデータを送る場合には、図 20 に示すように、先ず、ノード A からノード B へ、データの送信を要求する送信コマンドと、ノード B へ送信する送信データとを含むコマンドパケットを送信する。そして、このコマンドパケットを受信したノード B は、コマンドパケットに対するレスポンスとして、コマンドパケットを正常に受信したことを示す正常受信通知を含むレスポンスパケットをノード A へ送信する。

【0006】 また、例えば、ノード A からの要求によりノード B からノード A へデータを送る場合には、図 21 に示すように、先ず、ノード A からノード B へ、データの受信を要求する受信コマンドを含むコマンドパケットを送信する。そして、コマンドパケットを受信したノード B は、コマンドパケットに対するレスポンスとして、コマンドパケットを正常に受信したことを示す正常受信通知と、受信コマンドによって要求された受信データとを含むレスポンスパケットをノード A へ送信する。

20

【0007】 ところで、図 20 及び図 21 に示した例は、コマンドパケット及びレスポンスパケットをそれぞれ 1 パケットで送れる場合の例であるが、送るデータ量が 1 パケットの最大サイズを超える場合には、データを複数のパケットに分割しなければならない。すなわち、1 つのパケットのサイズには限りがあり、1 つの固まりをなす大量のデータを送る場合には、データを伝送可能なサイズに分割して送らなければならない。

30

【0008】 例えば、ノード A からノード B へデータを n 個のパケットに分割して送る際は、図 22 に示すように、先ず、ノード A からノード B へ、送信コマンドと $1/n$ 送信データとを含むコマンドパケット c_1 を送信する。ここで、 $1/n$ 送信データとは、ノード B へ送るデータを n 個に分割したうちの 1 つである。そして、このコマンドパケット c_1 を受信したノード B は、コマンドパケット c_1 に対するレスポンスとして、コマンドパケット c_1 を正常に受信したことを示す正常受信通知を含むレスポンスパケット r_1 をノード A へ送信する。そして、このやり取りを n 回繰り返す。したがって、1 回のコマンドパケットとレスポンスパケットのやり取りに要する時間を t とすると、データを n 回に分割して送信する場合には、 $n \times t$ の時間を要することになる。

40

【0009】

【発明が解決しようとする課題】 従来の Asynchronous モードでは、1 コマンドパケットを送る毎に 1 レスポンスパケットを受け取る必要があり、データを複数のパケットに分割して連続的に送った場合には、データの連続性を知る方法がない。このため、データを複数のパケットに分割して連続的に送るようなことは出来なかった。また、従来の Asynchronous モードでは、1 コマンドパケットを送る毎に 1 レスポンスパケットを受け取ることを想

50

定しており、送受信を行うノード間で複数のパケットにわたるデータ伝送の手順を確立する方法がなかった。すなわち、従来のAsynchronousモードでは、送受信ノード間でデータ伝送を行う手順を確立して連続してデータの送受信を行うというようなことは出来なかった。

【0010】すなわち、従来のAsynchronousモードにおけるコマンドパケットとレスポンスパケットのやり取りでは、それぞれが1対1の場合においてのみ対応づけが可能であり、複数のコマンドパケットや複数のレスポンスパケットに対する対応づけを行う方法がなかった。このため、従来のAsynchronousモードでは、1つの固まりをなす大量のデータを伝送可能なサイズに分割して送る場合にも、図22に示したように、1コマンドパケットを送る毎に1レスポンスパケットを受け取る必要があり、データ伝送に非常に長い時間が必要となっていた。換言すれば、従来のAsynchronousモードは、1パケットに納まるようなコマンドの送受信には好適であるが、1パケットに納まりきらないようなデータの送受信を行うには、非常に不都合が多かった。

【0011】なお、ここではIEEE1394バスを例に挙げたが、以上のような課題はIEEE1394バスに限らない。一般に、コマンドを送ることを想定しているパケットでは、1コマンドパケットを送る毎に1レスポンスパケットを受け取るようになされている。上述の課題は、このようなインターフェースに共通の課題である。

【0012】本発明は、以上のような従来の実情に鑑みて提案されたものであり、コマンドの送受信だけでなく、1パケットに納まりきらないようなデータの送受信も、コマンドの送受信に使用されるものと同様なパケットによって効率良く行うことが可能なデータ送信装置、データ受信装置、データ伝送システム及びデータ伝送方法を提供することを目的としている。

【0013】

【課題を解決するための手段】本発明に係るデータ送信装置は、所定のフォーマットのフレームを含むパケットによってデータを送信するデータ送信装置である。そして、上記フレームは、当該フレームのフォーマットの種別を示す情報が格納されるフィールドを有しており、このフォーマットの一つとして、送信データを複数のパケットに分割して送信する際にそれらのパケットの関係を示す情報が格納されるパケットラベルフィールドと、送信先機器に対するコマンド及び／又は任意のデータが格納されるデータフィールドとを上記フレームが有するように規定するフォーマットを備えていることを特徴とする。

【0014】また、本発明に係るデータ受信装置は、受信したパケットに応じて、所定のフォーマットのフレームを含むパケットによって返信データを返すデータ受信装置である。そして、上記フレームは、当該フレームの

フォーマットの種別を示す情報が格納されるフィールドを有しており、このフォーマットの一つとして、返信データを複数のパケットに分割して送信する際にそれらのパケットの関係を示す情報が格納されるパケットラベルフィールドと、受信したパケットに対するレスポンス及び／又は任意のデータが格納されるデータフィールドとを上記フレームが有するように規定するフォーマットを備えていることを特徴とする。

【0015】また、本発明に係るデータ送受信システムは、所定のフォーマットのフレームを含む送信用パケットによってデータを送信するデータ送信装置と、上記送信用パケットを受信したときに、当該送信用パケットによって送信された送信データに応じた返信データを、所定のフォーマットのフレームを含む返信用パケットによって上記データ送信装置へ送信するデータ受信装置とを有している。そして、上記送信用パケットに含まれるフレームは、当該フレームのフォーマットの種別を示す情報が格納されるフィールドを有しており、このフォーマットの一つとして、送信データを複数のパケットに分割して送信する際にそれらのパケットの関係を示す情報が格納されるパケットラベルフィールドと、上記データ受信装置に対するコマンド及び／又は任意のデータが格納されるデータフィールドとを当該フレームが有するように規定するフォーマットを備えていることを特徴とする。

【0016】また、本発明に係るデータ伝送方法は、所定のフォーマットのフレームを含む送信用パケットによって送信側ノードから受信側ノードへ送信データを送るとともに、上記送信用パケットを受信側ノードが受け取ったら、所定のフォーマットのフレームを含む返信用パケットによって受信側ノードから送信側ノードへ返信データを送ることにより、ノード間でデータの伝送を行うデータ伝送方法である。そして、上記送信用パケットに含まれるフレームは、当該フレームのフォーマットの種別を示す情報が格納されるフィールドを有しており、このフォーマットの一つとして、送信データを複数のパケットに分割して送信する際にそれらのパケットの関係を示す情報が格納されるパケットラベルフィールドと、受信側ノードに対するコマンド及び／又は任意のデータが格納されるデータフィールドとを当該フレームが有するように規定するフォーマットを備えていることを特徴とする。

【0017】

【発明の実施の形態】以下、本発明の実施の形態について、図面を参照しながら詳細に説明する。

【0018】図1に本発明の実施の形態の一例を示す。図1は、パーソナルコンピュータ1と、プリンタ2、デジタルスチルカメラ3と、セットトップボックス4とが相互にIEEE1394バス5によって接続された状態を示している。本発明は、図1に示すようにIEEE1

394バスによって接続されている機器間で、非同期なパケット送信モードであるAsynchronousモードによりデータの送受信を行うシステムに適用される。なお、Asynchronousモードでやり取りされるパケットは、Asynchronousパケットと称される。

【0019】Asynchronousパケットのやり取りを行う各機器は、IEEE1394バスによって接続可能な機器であればよい。このような機器としては、図1に示したパーソナルコンピュータ1、プリンタ2、デジタルスチルカメラ3及びセットトップボックス4の他にも、例えば、ハードディスク装置等の外部記憶装置や、プロッタ等の出力装置や、デジタルビデオディスクプレーヤ等のAV機器などが挙げられる。

【0020】従来、各機器間でAsynchronousパケットをやり取りする際は、1コマンドパケットに対して必ず1レスポンスパケットが対応していた。これに対して、本発明では、Asynchronousパケットをやり取りする際にも、図2に示すように1つのコマンドパケットに対して1つのレスポンスパケットが対応する場合と、図3に示すように複数のコマンドパケットに対して1つのレスポンスパケットが対応する場合と、図4に示すように1つのコマンドパケットに対して複数のレスポンスパケットが対応する場合とが可能とされる。

【0021】複数のコマンドパケットに対して1つのレスポンスパケットが対応する場合の一例を図5に示す。図5は、ノードBに書き込むデータをn個のパケットに分割してノードAから送信するときのパケットのやり取りの手順を示している。このときは、まず、書き込みコマンドと1/nデータとを含むコマンドパケットを、ノードAからノードBへ送信する。ここで、書き込みコマンドとは、ノードBへのデータの書き込みを要求するコマンドであり、1/nデータとは、ノードBに書き込むデータをn個に分割したうちのひとつである。次に、同様に、書き込みコマンドと次の1/nデータとを含むコマンドパケットを、ノードAからノードBへ送信する。そして、このようなコマンドパケットのノードAからノードBへの送信をn回繰り返す。そして、ノードBは、n個のコマンドパケットを正常に受信したら、それらのコマンドパケットに対するレスポンスとして、それらのコマンドパケットを正常に受信したことを示す正常受信通知を含むレスポンスパケットをノードAへ送信する。以上のようなパケットのやり取りにより、n個のパケットに分割されてノードAから送信されたデータがノードBに書き込まれることとなる。

【0022】また、1つのコマンドパケットに対して複数のレスポンスパケットが対応する場合の一例を図6に示す。図6は、ノードBから読み出すデータをn個のパケットに分割してノードAへ送信するときのパケットのやり取りの手順を示している。このときは、まず、読み出しコマンドを含むコマンドパケットを、ノードAから

ノードBへ送信する。ここで、読み出しコマンドは、ノードBからのデータの読み出しを要求するコマンドである。そして、読み出しコマンドを含むコマンドパケットを受け取ったノードBは、1/n読み出しデータを含むレスポンスパケットをノードAへ送信する。ここで、1/n読み出しデータとは、ノードBから読み出すデータをn個に分割したうちのひとつである。そして、このようなレスポンスパケットのノードBからノードAへの送信をn回繰り返す。以上のようなパケットのやり取りにより、ノードBから読み出されたデータがn個のパケットに分割されて、ノードAへ送信されることとなる。

【0023】本発明では、図5に示したように、複数のコマンドパケットに対して1つのレスポンスパケットが対応するようにしたり、或いは、図6に示したように、1つのコマンドパケットに対して複数のレスポンスパケットが対応するようにしたりすることを可能とする。これにより、データを複数のパケットに分割して送信するようなときに、1コマンドパケットに対して必ず1レスポンスパケットを返すような方法に比べて、パケットのやり取りを大幅に少なくすることができ、データ伝送に要する時間を大幅に短縮することができる。

【0024】つぎに、以上のような伝送方式を可能にするパケットのフレーム構造について説明する。

【0025】IEEE1394で規定されているAsynchronousパケットは、図7に示すようなフレーム構造を持つ。すなわち、Asynchronousパケットは、当該パケットについての情報が記述されるパケットヘッダと、送信データがのせられるデータブロックとを有している。

【0026】パケットヘッダには、destination_ID、tl(transact label)、rt(retry code)、tcode、pri(priority)、source_ID、destination_offset、data_length、extended_tcode及びheader_CRCの各フィールドがある。

【0027】ここで、destination_IDフィールドには、パケット送信先のノードのID、具体的にはIEEE1394バスで規定されたノードのID番号(NodeID)が記述される。tlフィールドには、パケットの番号、具体的には、パケットの送信先及び送信元のノードが自らに關係するトランザクションであることを認識できるようにするための固有の値が記述される。rtフィールドには、ビジー時のリトライ方法に関する情報が記述される。tcodeフィールドには、コマンドレジスタ又はレスポンスレジスタにメッセージを書き込むことを指示するコードが記述される。

【0028】また、priフィールドには、パケットの優先順位が記述される。source_IDフィールドには、パケット送信元のノードのID、具体的には、IEEE1394バスで規定されたノードのID番号(NodeID)が記述される。destination_offsetフィールドには、コマンドレジスタ又はレスポンスレジスタのアドレスが記述される。data_lengthフィールドには、パケットヘッダに

続くデータブロックのデータ長に関する情報が記述される。extended_tcodeフィールドは、tcodeを拡張する場合に利用される。header_CRCフィールドには、パケットヘッダのチェックサムを行うCRC計算値が記述される。

【0029】一方、データブロックには、cts(command and Transaction Set)、FCPdata(Function Control Protocol data)及びdata_CRCの各フィールドがある。ここで、ctsフィールド及びFCPdataフィールドの部分は、FCPフレームと称される。また、data_CRCフィールドには、データブロックのチェックサムを行うCRC計算値が記述される。

【0030】ctsフィールドは、FCPフレームのフォーマットを定義する。すなわち、Asynchronousパケットは、FCPフレームを含んでおり、FCPフレームは、当該FCPフレームのフォーマットの種別を示す情報が格納されるフィールドとして、ctsフィールドを有している。

【0031】FCPフレームについては、IEC1883(International Electrotechnical Commission 1883)によって定義されている。そして、FCPフレームのフォーマットの種別は、ctsフィールドの値によって分類される。例えば、送信されるパケットがコマンドパケットのときに、ctsフィールドの値が0000ならば、そのコマンドパケットのFCPフレームは、IEEE1394のAV/C Digital Interface Command Setで規定されたコマンドフレーム(以下、AV/Cコマンドフレームと称する。)である。また、送信されるパケットがレスポンスパケットのときに、ctsフィールドの値が0000ならば、そのレスポンスパケットのFCPフレームは、IEEE1394のAV/C Digital Interface Command Setで規定されたレスポンスフレーム(以下、AV/Cレスポンスフレームと称する。)である。

【0032】AV/Cコマンドフレームは、図8に示すようなフレーム構造とされる。すなわち、AV/Cコマンドフレームには、cts、ctype、subunit_type、subunit_ID、opcode、operandの各フィールドがある。なお、Asynchronousパケットは32bits単位とされるので、AV/Cコマンドフレームには、その合計が32bitsの整数倍となるように、必要に応じて、operandフィールドの後に0が付加される。

【0033】ctsフィールドは、上述したように、AV/Cコマンドフレームのときには、0000とされる。ctypeフィールドには、コマンドの種類が記述される。例えば、ctypeの値が0000のときには、当該パケットによるコマンドが、パケット送信先機器の制御を行うコマンド、すなわちCONTROLコマンドであることを表す。また、例えば、ctypeの値が0001のときには、当該パケットによるコマンドが、パケット送信先機器の状態を問い合わせるコマンド、すなわちSTATUSコマンド

であることを表す。

【0034】subunit_typeフィールドには、コマンドが適用されるノードのタイプが記述される。例えば、subunit_typeフィールドの値が0000のときには、コマンドが適用されるノードがモニタであることを表す。また、例えば、subunit_typeフィールドの値が00100のときには、コマンドが適用されるノードがビデオカセットレコーダであることを表す。subunit_IDフィールドには、当該パケットによるコマンドが適用される機器のID番号が記述される。

【0035】opcodeフィールドには、subunit_typeフィールドで示されたノードに対する具体的なコマンドのコード、すなわちオペレーションコードが記述される。例えば、subunit_typeフィールドの値が00100であり、opcodeフィールドの値がC3hのとき、当該パケットによるコマンドは「ビデオカセットレコーダ」の「再生」である。また、subunit_typeフィールドの値が00100であり、opcodeフィールドの値がC2hのとき、当該パケットによるコマンドは「ビデオカセットレコーダ」の「録画」である。

【0036】operandフィールドは、Asynchronousパケットの大きさが最大値を越えない範囲であれば、任意の数のフィールドを設定することができる。そして、これらのフィールドoperand[0], operand[1], ..., operand[n]には、opcodeフィールドで設定されたオペレーションコードを実行する上で必要とする情報が、必要に応じて記述される。具体的には、例えば、再生の速度や方向等についての情報が記述される。

【0037】一方、AV/Cレスポンスフレームは、図9に示すようなフレーム構造とされる。図9に示すように、AV/CレスポンスフレームはAV/Cコマンドフレームとほぼ同様なフレーム構造とされるが、AV/Cコマンドフレームにおけるctypeフィールドの部分が、responseフィールドとされる。responseフィールドには、コマンドに対するレスポンスが記述される。すなわち、responseフィールドには、例えば、コマンドの受信が正常に完了したときに、受信が正常に完了したことを示す値が記述される。

【0038】以上、図7乃至図9に示したフレーム構造は、本出願時にIEEE1394並びにIEC1883で規定されている内容であり、その詳細については、それらの規格を参照されたい。そして、本発明では、以上のようなフレーム構造を拡張して、FCPフレームのフォーマットの一つとして、図10及び図11に示すようなフォーマットを追加する。

【0039】図10は、本発明を適用する際に使用されるコマンドフレーム(以下、新規コマンドフレームと称する。)の一例を示している。すなわち、本発明をIEEE1394バスに適用する際は、コマンドパケットのctsフィールドによって規定されるFCPフレームのフ

フォーマットのの一つとして、図10に示すようなフレーム構造を定義する。この新規コマンドフレームを採用することにより、単にコマンドを送るだけでなく、任意の送信データを複数のパケットに分割して送信することが可能となる。

【0040】また、図11は、本発明を適用する際に使用されるレスポンスフレーム（以下、新規レスポンスフレームと称する。）の一例を示している。すなわち、本発明をIEEE1394バスに適用する際は、レスポンスパケットのctsフィールドによって規定されるFCPフレームのフォーマットの一つとして、図11に示すようなフレーム構造を定義する。この新規レスポンスフレームを採用することにより、単にレスポンスを返すだけでなく、任意の返信データを複数のパケットに分割して送信することが可能となる。

【0041】なお、新規コマンドフレームや新規レスポンスフレームを用いて送受信されるデータは任意であるが、あえて例示するならば、例えば、動画データ、静止画像データ、音声データ、文字データ、プログラムモジュール等が挙げられる。

【0042】図10に示す新規コマンドフレームは、AV/Cコマンドフレームに対して、c(Continuity)、cl(command label)及びsub-l(sub_label)の各フィールドを追加したものであり、cフィールド、clフィールド及びsub-lフィールド以外については、AV/Cコマンドフレームと同様である。ただし、新規コマンドフレームは、ctsフィールドの値によってAV/Cコマンドフレームとは別のフレームとして区別されるので、コマンドとコードの対応等については、AV/Cコマンドフレームの場合と同一でなくてもよい。

【0043】新規コマンドフレームにおいて、clフィールド及びsub-lフィールドには、送信データを複数のパケットに分割して送信する際に、それらのパケットの関係を示す情報が格納される。具体的には、clフィールドには、送信データを複数のパケットに分割して送信する際に、それらのパケットに共通の値が設定され、sub-lフィールドには、送信データを複数のパケットに分割して送信する際に、それらのパケットの順序を示す情報が格納される。また、clフィールドには、送信データを複数のパケットに分割して送信する際に、最後のパケットであるか否かを示す情報が格納される。

【0044】換言すれば、clフィールドは、各コマンドパケットに対するラベル番号を示しており、sub-lフィールドは、同一ラベル番号をもつ複数のコマンドパケットに対するサブラベル番号を示しており、cフィールドは、同一ラベル番号を持つ複数のコマンドパケットが続いて送られるか否かを示している。

【0045】コマンドパケットにclフィールドを設けることにより、送信データを複数のコマンドパケットに分割しても、それらのコマンドパケットを対応づけること

が可能となる。また、コマンドパケットにsub_lフィールドを設けることにより、同一のラベル番号を持つコマンドパケット、すなわちclフィールドの値が同じコマンドパケットの順序を知ることが可能となる。これにより、データが複数のコマンドパケットに分割されていても、それらのコマンドパケットの順序を知ることができ、更にそれら一連のコマンドパケットに抜けがないかを確認することもできる。

【0046】以上のような新規コマンドフレームを用いて、コマンドを送信する際、operandフィールドの使い方は、AV/Cコマンドフレームのときと同様である。一方、新規コマンドフレームで任意のデータを送信する際は、AV/Cコマンドフレームにおけるoperandフィールドに相当する部分に送信データをのせる。すなわち、この新規コマンドフレームは、送信先ノードに対するコマンド及び/又は任意の送信データが格納されるデータフィールドを備えている。したがって、この新規コマンドフレームを用いることにより、単にコマンドを送るだけでなく、任意のデータを送ることも可能となる。

【0047】一方、図11に示す新規レスポンスフレームは、AV/Cレスポンスフレームに対して、c(Continuity)、rl(response label)及びsub-l(sub_label)の各フィールドを追加したものであり、cフィールド、rlフィールド及びsub-lフィールド以外については、AV/Cレスポンスフレームと同様である。ただし、新規レスポンスフレームは、ctsフィールドの値によってAV/Cレスポンスフレームとは別のフレームとして区別されるので、レスポンスとコードの対応等については、AV/Cレスポンスフレームの場合と同一でなくてもよい。

【0048】新規レスポンスフレームにおいて、rlフィールド及びsub-lフィールドには、返信データを複数のパケットに分割して送信する際に、それらのパケットの関係を示す情報が格納される。具体的には、rlフィールドには、返信データを複数のパケットに分割して送信する際に、それらのパケットに共通の値が設定され、sub-lフィールドには、返信データを複数のパケットに分割して送信する際に、それらのパケットの順序を示す情報が格納される。また、rlフィールドには、返信データを複数のパケットに分割して送信する際に、最後のパケットであるか否かを示す情報が格納される。

【0049】換言すれば、rlフィールドは、各レスポンスパケットに対するラベル番号を示しており、sub-lフィールドは、同一ラベル番号をもつ複数のレスポンスパケットに対するサブラベル番号を示しており、cフィールドは、同一ラベル番号を持つ複数のレスポンスパケットが続いて送られるか否かを示している。

【0050】レスポンスパケットにrlフィールドを設けることにより、返信データを複数のレスポンスパケットに分割しても、それらのレスポンスパケットを対応づけることが可能となる。また、レスポンスパケットにsub_

1フィールドを設けることにより、同一のラベル番号を持つレスポンスパケット、すなわちr1フィールドの値が同じレスポンスパケットの順序を知ることが可能となる。これにより、データが複数のレスポンスパケットに分割されていても、それらのレスポンスパケットの順序を知ることができ、更にそれら一連のレスポンスパケットに抜けがないかを確認することもできる。

【0051】以上のような新規レスポンスフレームによって、単なるレスポンスを返すような場合、operandフィールドの使い方は、AV/Cレスポンスフレームのときと同様である。一方、新規レスポンスフレームで単なるレスポンスを返すだけでなく任意の返信データを送信するような場合には、AV/Cレスポンスフレームにおけるoperandフィールドに相当する部分に返信データをのせる。すなわち、この新規レスポンスフレームは、受信したパケットに対するレスポンス及び／又は任意の返信データが格納されるデータフィールドを備えている。したがって、この新規レスポンスフレームを用いることにより、受信したパケットに対して単なるレスポンスを返すだけでなく、受信したパケットに応じて任意のデータを返すことも可能となる。

【0052】ところで、図10に示した例では、AV/Cコマンドフレームにフィールドを追加して新規コマンドフレームとした。また、図11に示した例では、AV/Cレスポンスフレームにフィールドを追加して新規レスポンスフレームとした。しかし、本発明を適用するにあたって新規に定義するコマンドフレームやレスポンスフレームは、AV/CコマンドフレームやAV/Cレスポンスフレームで使用していたフィールドの全てが必要であるとは限らない。

【0053】すなわち、例えば図12に示すように、本発明を適用するにあたって新規に定義するコマンドフレームとして、subunit_typeフィールドやsubunit_IDフィールドを持たないようなフレーム構造を定義するようにしても良いし、また、例えば図13に示すように、本発明を適用するにあたって新規に定義するレスポンスフレームとして、subunit_typeフィールドやsubunit_IDフィールドを持たないようなフレーム構造を定義するようにしても良い。

【0054】つぎに、図10に示した新規コマンドフレームを有するコマンドパケットと、図11に示した新規レスポンスフレームを有するレスポンスパケットとを用いたときのパケットのやり取りについて、図14及び図15を参照して説明する。

【0055】図14は、複数のコマンドパケットに対して1つのレスポンスパケットを返すときのパケットのやり取りの一例として、送信データを3つに分割して、第1乃至第3のコマンドパケットc1-1, c1-2, c1-3によってノードAからノードBへ送信する様子を示している。このときは、ノードAからノードBへ3つのコマンドパ

ットc1-1, c1-2, c1-3を送り、それらのコマンドパケットc1-1, c1-2, c1-3に対して、ノードBからノードAへ1つのレスポンスパケットr1-1が返される。

【0056】図14に示すように、送信データを3つのコマンドパケットc1-1, c1-2, c1-3に分割して送信する際に、それらのコマンドパケットc1-1, c1-2, c1-3において、c1フィールドには、それらのコマンドパケットc1-1, c1-2, c1-3に対するラベル番号が設定される。具体的には、図14の例では、第1乃至第3のコマンドパケットc1-1, c1-2, c1-3におけるc1フィールドの値を全て「1」としている。このように、c1フィールドには、それらの3つのコマンドパケットc1-1, c1-2, c1-3が一連のコマンドパケットであることを示すように、それらのパケットに共通の値が設定される。

【0057】また、sub_1フィールドには、それらの3つのコマンドパケットc1-1, c1-2, c1-3の順序を示す情報が格納される。具体的には、図14の例では、3つに分割された送信データの最初の部分を含む第1のコマンドパケットc1-1では、sub_1フィールドに、先頭のコマンドパケットであることを示す「0」が設定される。また、送信データの次の部分を含む第2のコマンドパケットc1-2では、sub_1フィールドに、2番目のコマンドパケットであることを示す「1」が設定される。また、送信データの最後の部分を含む第3のコマンドパケットc1-3では、sub_1フィールドに、3番目のコマンドパケットであることを示す「2」が設定される。

【0058】また、cフィールドには、最後のコマンドパケットであるか否かを示す情報が格納される。具体的には、図14の例では、第1及び第2のコマンドパケットc1-1, c1-2では、同じラベル番号を持つコマンドパケットが、これらのコマンドパケット以降に存在していることを示す「1」がcフィールドに設定される。一方、第3のコマンドパケットc1-3では、同じラベル番号を持つコマンドパケットが、このコマンドパケット以降に無いことを示す「0」が設定される。

【0059】これら3つのコマンドパケットc1-1, c1-2, c1-3を受け取ったノードBは、c1フィールド、sub_1フィールド及びcフィールドに記述された内容に基づいて、それらのコマンドパケットc1-1, c1-2, c1-3から、3つに分割される前の送信データを復元する。

【0060】そして、cフィールドに「0」が設定されたコマンドパケット、すなわち最後のコマンドパケットc1-3を受け取ったノードBは、レスポンスパケットr1-1をノードAへ送信する。このときは、r1フィールドには、コマンドパケットc1-1, c1-2, c1-3のc1フィールドに設定されていたものと同じラベル番号、すなわち「1」が設定される。また、sub_1フィールドには、先頭のレスポンスパケットであることを示す「0」が設定される。また、cフィールドには、同じラベル番号を持つレスポンスパケットが、このレスポンスパケット以降に無

いことを示す「0」が設定される。

【0061】以上のようにして、3つのコマンドパケットc1-1, c1-2, c1-3によって、送信データがノードAからノードBへと送られる。

【0062】一方、図15は、1つのコマンドパケットに対して複数のレスポンスパケットを返すときのパケットのやり取りの一例として、ノードAからノードBへのコマンドパケットに対する返信データを3つに分割して、第1乃至第3のレスポンスパケットr2-1, r2-2, r2-3によってノードBからノードAへ送信する様子を示している。このときは、ノードAからノードBへ1つのコマンドパケットc2-1を送り、そのコマンドパケットc2-1に対してノードBからノードAへ3つのレスポンスパケットr2-1, r2-2, r2-3が返される。

【0063】図15に示す例では、まず、ノードAからノードBへコマンドパケットc2-1が送信される。このコマンドパケットc2-1において、c1フィールドには、当該コマンドパケットc2-1に対するラベル番号、例えば「1」が設定される。また、sub_1フィールドには、先頭のコマンドパケットであることを示す「0」が設定される。また、cフィールドには、同じラベル番号を持つコマンドパケットが、このコマンドパケット以降に無いことを示す「0」が設定される。そして、図15に示す例では、このコマンドパケットc2-1に対する返信データを3つに分割して、ノードBからノードAへ送信している。

【0064】このように返信データを3つのレスポンスパケットr2-1, r2-2, r2-3に分割して送信する際、それらのレスポンスパケットr2-1, r2-2, r2-3において、r1フィールドには、コマンドパケットc2-1のc1フィールドに設定されていたものと同じラベル番号が設定される。具体的には、図15の例では、第1乃至第3のレスポンスパケットr2-1, r2-2, r2-3におけるr1フィールドの値が全て「1」とされる。このように、r1フィールドには、それらの3つのレスポンスパケットr2-1, r2-2, r2-3が一連のレスポンスパケットがあることを示すように、それらのパケットに共通の値が設定される。

【0065】また、sub_1フィールドには、それらの3つのレスポンスパケットr2-1, r2-2, r2-3の順序を示す情報が格納される。具体的には、図15の例において、3つに分割された返信データの最初の部分を含む第1のレスポンスパケットr2-1には、sub_1フィールドに、先頭のレスポンスパケットであることを示す「0」が設定される。また、返信データの次の部分を含む第2のレスポンスパケットr2-2には、sub_1フィールドに、2番目のレスポンスパケットであることを示す「1」が設定される。また、返信データの最後の部分を含む第3のレスポンスパケットr2-3には、sub_1フィールドに、3番目のレスポンスパケットであることを示す「2」が設定される。

【0066】また、cフィールドには、最後のレスポンスパケットであるか否かを示す情報が格納される。具体的には、図15の例では、第1及び第2のレスポンスパケットr2-1, r2-2では、同じラベル番号を持つレスポンスパケットが、これらのレスポンスパケット以降に存在していることを示す「1」がcフィールドに設定される。一方、第3のレスポンスパケットr2-3では、同じラベル番号を持つレスポンスパケットが、このレスポンスパケット以降に無いことを示す「0」が設定される。

10 【0067】そして、これら3つのレスポンスパケットr2-1, r2-2, r2-3を受け取ったノードAは、r1フィールド、sub_1フィールド及びcフィールドに記述された内容に基づいて、それらのレスポンスパケットr2-1, r2-2, r2-3から、3つに分割される前の返信データを復元する。

【0068】以上のようにして、3つのレスポンスパケットr2-1, r2-2, r2-3によって、返信データがノードBからノードAへと送られる。

20 【0069】つぎに、新規コマンドフレームを有するコマンドパケット、及び新規レスポンスフレームを有するレスポンスパケットを用いてデータ伝送を行う際の手順の一例について、図16に示すデータ伝送概念モデルを参照して詳細に説明する。

【0070】図16に示すように、新規コマンドフレームを有するコマンドパケット、及び新規レスポンスフレームを有するレスポンスパケットを用いてデータ伝送を行う際は、まず、シリアルバスでつながれた2つのノードA, Bにおいて、相手先を特定するために仮想の接続10を張る。このとき、接続10にはID番号が割り当てられる。このID番号のことをConnectIDと称する。そして、接続10を張った各ノードA, Bは、ConnectIDにより接続10の相手先を識別する。

30 【0071】なお、IEEE1394バスでは、各ノードの識別のために各ノードにID番号(NodeID)を割り当てるが、バスリセットが起こると、各ノードのID番号が変わってしまうことがある。このようなときには、ConnectIDをもとに再び接続10を張り直す。すなわち、各接続10にConnectIDを割り付けることにより、バスリセットによりノードのID番号が変わっても、再び接続10を張り直すことが可能となる。

【0072】また、1つの接続10の中には、更に仮想的なチャンネル11, 12を設け、それらのチャンネル11, 12にID番号を割り当てる。このID番号のことをChannelIDと称する。ここで、1つの接続10の中には、必要に応じて複数のチャンネルを設けることができる。

50 【0073】そして、接続10に割り当てられたConnectIDと、各チャンネル11, 12に割り当てられたChannelIDとに基づいて、ノードAとノードBとの間

で各チャンネル11, 12ごとにコマンドやデータのやり取りを行う。このように、1コネクション内に更に仮想的なチャンネルを設けることにより、1つのコネクションで複数のコマンドやデータのやり取りが可能となる。

【0074】つぎに、以上のようなデータ伝送に使用されるコマンドについて説明する。本例では、以上のようなデータ伝送を実現するために、CONNECTコマンドと、DISCONNECTコマンドと、OPENコマンドと、CLOSEコマンドと、READコマンドと、WRITEコマンドと、ABORTコマンドとを規定する。

【0075】CONNECTコマンド

CONNECTコマンドは、あるノードが別のノードに対してデータやコマンドを伝送するための仮想的なコネクションを張るときに用いる。具体的には、あるノードから別のノードに対してコネクションを張ろうとするときに、ctypeフィールドにCONTROLコマンドを設定し、且つ、opcodeフィールドにCONNECTコマンドを設定したコマンドパケットを、コネクションを張ろうとする相手先のノードへ送信する。

【0076】また、CONNECTコマンドは、相手先のノードに対して既にコネクションを張られているかどうかを知るようなときにも使用される。具体的には、相手先のノードのコネクションの状態を知りたいときに、ctypeフィールドにSTATUSコマンドを設定し、且つ、opcodeフィールドにCONNECTコマンドを設定したコマンドパケットを、相手先のノードへ送信する。

【0077】DISCONNECTコマンド

DISCONNECTコマンドは、ノードとノードとの間で張られていたコネクションを開放するときに用いる。具体的には、コネクションを開放するときに、ctypeフィールドにCONTROLコマンドを設定し、且つ、opcodeフィールドにDISCONNECTコマンドを設定したコマンドパケットを、コネクションの相手先のノードへ送信する。

【0078】OPENコマンド

OPENコマンドは、あるノードが別のノードに対して仮想的なチャンネルを確立するときに用いる。具体的には、あるノードから別のノードに対してチャンネルを確立しようとするときに、ctypeフィールドにCONTROLコマンドを設定し、且つ、opcodeフィールドにOPENコマンドを設定したコマンドパケットを、チャンネルを確立しようとする相手先のノードへ送信する。

【0079】また、OPENコマンドは、既に確立されているチャンネルのChannelIDを問い合わせるときにも使用される。具体的には、既に確立されているチャンネルのChannelIDを知りたいときには、ctypeフィールドにSTATUSコマンドを設定し、且つ、opcodeフィールドにOPENコマンドを設定したコマンドパケットを、相手先のノードへ送信する。

【0080】CLOSEコマンド

CLOSEコマンドは、ノードとノードとの間で確立してい

たチャンネルを開放するときに用いる。具体的には、チャンネルを開放するときに、ctypeフィールドにCONTROLコマンドを設定し、且つ、opcodeフィールドにCLOSEコマンドを設定したコマンドパケットを、開放しようとするチャンネルの相手先のノードへ送信する。

【0081】READコマンド

READコマンドは、あるノードが別のノードの持つデータを読み出すときに用いる。具体的には、別のノードの持つデータを読み出そうとするときに、ctypeフィールドにCONTROLコマンドを設定し、且つ、opcodeフィールドにREADコマンドを設定したコマンドパケットを、読み出したいデータを持つノードへ送信する。

【0082】また、READコマンドは、相手先ノードのコンテンツのうち読み出し可能なデータ量を問い合わせるときにも使用される。具体的には、読み出し可能なデータ量を問い合わせるときに、ctypeフィールドにSTATUSコマンドを設定し、且つ、opcodeフィールドにREADコマンドを設定したコマンドパケットを、問い合わせ先のノードへ送信する。

【0083】WRITEコマンド

WRITEコマンドは、あるノードが別のノードに対してあるコンテンツのデータを書き込むときに用いる。具体的には、別のノードに対してデータを書き込もうとするときに、ctypeフィールドにCONTROLコマンドを設定し、且つ、opcodeフィールドにWRITEコマンドを設定したコマンドパケットを、データ書き込み先のノードへ送信する。

【0084】また、WRITEコマンドは、相手先ノードの書き込み可能な空き領域量を問い合わせるときにも使用される。具体的には、書き込み可能な空き領域量を問い合わせるときに、ctypeフィールドにSTATUSコマンドを設定し、且つ、opcodeフィールドにWRITEコマンドを設定したコマンドパケットを、問い合わせ先のノードへ送信する。

【0085】つぎに、上述のようなコマンドを用いて、ノードAからノードBにデータを書き込む際の手順について、図17を参照して説明する。ここで、ノードBに書き込むデータは、複数のパケットに分割してノードAから送信するものとしている。

【0086】図17に示した手順では、まず、ノードAとノードBとの間でコネクションを張り、次に、そのコネクションの中でチャンネルを確立する。次いで、ノードBのデータ書き込み可能領域量を確認した上で、複数のコマンドパケットを用いてノードBへのデータの書き込みを行う。その後、チャンネルを開放し、更に、ノードAとノードBとの間のコネクションを開放する。以下、この手順について、図17を参照して詳細に説明する。

【0087】まず、ノードAがノードBに対して、CONNECTコマンドを用いて、コネクションを要求する。このとき、ノードAはノードBに対して、ノードAのID番

21

号NodeID(A)と、張ろうとするコネクションに対して割り当てたConnectIDとをoperandフィールドに記述したコマンドパケット21を送る。具体的には、表1に示すように各フィールドが設定されたコマンドパケット21 *

| フィールド* | cl | sub_l | c | ctype | opcode | operand[0] | operand[1] |
|--------|----|-------|---|---------|---------|------------|------------|
| 設定値 | 1 | 0 | 0 | CONTROL | CONNECT | ConnectID | NodeID(A) |

【0089】コマンドパケット21によりコネクションが正しく張れたら、次に、ノードBは、コネクションが正しく張れたことを示すレスポンスをノードAへ返す。このとき、ノードBはノードAに対して、ノードBのID番号NodeID(B)と、割り当てられたConnectIDとをoperandフィールドに記述したレスポンスパケット22を送 ※

| フィールド* | cl | sub_l | c | ctype | operand[0] | operand[1] |
|--------|----|-------|---|----------|------------|------------|
| 設定値 | 1 | 0 | 0 | ACCEPTED | ConnectID | NodeID(B) |

【0091】以上のやり取りにより、ノードAとノードBとの間でコネクションが張られる。なお、このやり取りは、ノードAとノードBとの間で既にコネクションが張られている場合には、改めて行う必要はない。

【0092】次に、ノードAはノードBに対して、OPENコマンドを用いてチャネルの確立を求める。このとき、ノードAはノードBに対して、使用するコネクションの☆

| フィールド* | cl | sub_l | c | ctype | opcode | operand[0] | operand[1] |
|--------|----|-------|---|---------|--------|------------|------------|
| 設定値 | 1 | 0 | 0 | CONTROL | OPEN | ConnectID | ChannelID |

【0094】コマンドパケット23によりチャネルが確立されたら、次に、ノードBは、チャネルが確立されたことを示すレスポンスをノードAへ返す。このとき、ノードBはノードAに対して、使用するコネクションのConnectIDと、割り当てられたChannelIDとをoperandフィールドに記述したレスポンスパケット24を送る。具体☆

| フィールド* | cl | sub_l | c | ctype | operand[0] | operand[1] |
|--------|----|-------|---|----------|------------|------------|
| 設定値 | 1 | 0 | 0 | ACCEPTED | ConnectID | ChannelID |

【0096】以上のやり取りにより、ノードAとノードBとの間でチャネルが確立される。なお、このやり取りは、ノードAとノードBとの間で既にチャネルが確立されている場合には、改めて行う必要はない。

【0097】次に、ノードAはノードBに対して、WRITEコマンドを用いて、ノードBの書き込み可能な空き領域を問い合わせる。このとき、ノードAはノードBに

22

*を、ノードAからノードBへ送出する。

【0088】

【表1】

※る。具体的には、表2に示すように各フィールドが設定されたレスポンスパケット22が、ノードBからノードAへ返される。

【0090】

【表2】

☆ConnectIDと、確立しようとするチャネルに対して割り当てたChannelIDとをoperandフィールドに記述したコマンドパケット23を送る。具体的には、表3に示すように各フィールドが設定されたコマンドパケット23を、ノードAからノードBへ送出する。

【0093】

【表3】

☆的には、表4に示すように各フィールドが設定されたレスポンスパケット24が、ノードBからノードAへ返される。

【0095】

【表4】

に対して、使用するコネクションのConnectIDと、使用するチャネルのChannelIDとをoperandフィールドに記述したコマンドパケット25を送る。具体的には、表5に示すように各フィールドが設定されたコマンドパケット25を、ノードAからノードBへ送出する。

【0098】

【表5】

| フィールド* | cl | sub_l | c | ctype | opcode | operand[0] | operand[1] |
|--------|----|-------|---|--------|--------|------------|------------|
| 設定値 | 1 | 0 | 0 | STATUS | WRITE | ConnectID | ChannelID |

【0099】次に、コマンドパケット25を受け取ったノードBは、ノードBの書き込み可能な空き領域量をノードAに知らせる。このとき、ノードBはノードAに対して、使用する接続のConnectIDと、使用するチャンネルのChannelIDと、ノードBの書き込み可能な空き領域量AvailableBufferSizeとをoperandフィールドに*

*記述したレスポンスパケット26を送る。具体的には、表6に示すように各フィールドが設定されたレスポンスパケット26を、ノードBからノードAへ送出する。

【0100】

【表6】

| フィールド* | cl | sub_l | c | ctype | operand[0] | operand[1] | operand[2] |
|--------|----|-------|---|----------|------------|------------|---------------------|
| 設定値 | 1 | 0 | 0 | ACCEPTED | ConnectID | ChannelID | AvailableBufferSize |

【0101】次に、ノードAはノードBに対して、WRITEコマンドを用いて、ノードBにデータを書き込む。ここで、ノードBに書き込むデータは、n個に分割されており、先ず、最初のデータであるWritingData1を送信する。このとき、ノードAはノードBに対して、使用する接続のConnectIDと、使用するチャンネルのChannelIDとをoperandフィールドに記述するとともに、FCPda※

※taフィールドの残りの領域にWritingData1が記述されたコマンドパケット27-1を送る。具体的には、表7に示すように各フィールドが設定されたコマンドパケット27-1を、ノードAからノードBへ送出する。

【0102】

【表7】

| フィールド* | cl | sub_l | c | ctype | opcode | operand[0] | operand[1] | |
|--------|----|-------|---|---------|--------|------------|------------|--------------|
| 設定値 | 1 | 0 | 1 | CONTROL | WRITE | ConnectID | ChannelID | WritingData1 |

【0103】その後、同様に、ノードAはノードBに対してn個に分割されたデータの送信を順次行う。このとき、sub_lフィールドの値は、データを分割して送信する各コマンドパケットの順序を示すように、各コマンドパケット毎に1ずつ増加される。

【0104】そして、最後に、n個に分割されたデータのうちの最後のデータであるWritingDataNを送信する。このとき、ノードAはノードBに対して、使用する接続のConnectIDと、使用するチャンネルのChannelIDとをoperandフィールドに記述するとともに、FCPdataフィールドの残りの領域にWritingDataNを記述したコマンド★

★ドパケットを送る。具体的には、表8に示すように各フィールドが設定されたコマンドパケット27-nを、ノードAからノードBへ送出する。このとき、cフィールドの値は、コマンドパケット27-nの前のコマンドパケットまでは「1」とされるが、コマンドパケット27-nでは、このコマンドパケット27-nがn個に分割されたデータの送信を行うパケットうちの最後のパケットであることを示すように「0」とされる。

【0105】

【表8】

| フィールド* | cl | sub_l | c | ctype | opcode | operand[0] | operand[1] | |
|--------|----|-------|---|---------|--------|------------|------------|--------------|
| 設定値 | 1 | n | 0 | CONTROL | WRITE | ConnectID | ChannelID | WritingDataN |

【0106】以上のコマンドパケット27-1～27-nによって送られてきたデータが、ノードBに正しく書き込まれたら、次に、ノードBは、データが正しく書き込まれたことを示すレスポンスをノードAへ返す。このとき、ノードBはノードAに対して、使用する接続のConnectIDと、使用するチャンネルのChannelIDとを

operandフィールドに記述したレスポンスパケット28を送る。具体的には、表9に示すように各フィールドが設定されたレスポンスパケット28が、ノードBからノードAへ返される。

【0107】

【表9】

25

26

| フィールド | cl | sub_l | c | ctype | operand[0] | operand[1] |
|-------|----|-------|---|----------|------------|------------|
| 設定値 | 1 | 0 | 0 | ACCEPTED | ConnectID | ChannelID |

【0108】次に、ノードAはノードBに対して、CLOSEコマンドを用いて、チャンネルの開放を要求する。このとき、ノードAはノードBに対して、使用する接続のConnectIDと、開放しようとするチャンネルのChannelIDとをoperandフィールドに記述したコマンドパケット10

*ト29を送る。具体的には、表10に示すように各フィールドが設定されたコマンドパケット29を、ノードAからノードBへ送出する。

【0109】

【表10】

| フィールド | cl | sub_l | c | ctype | opcode | operand[0] | operand[1] |
|-------|----|-------|---|---------|--------|------------|------------|
| 設定値 | 1 | 0 | 0 | CONTROL | CLOSE | ConnectID | ChannelID |

【0110】コマンドパケット29によりチャンネルの開放がなされたら、次に、ノードBは、チャンネルが開放されたことを示すレスポンスをノードAへ返す。このとき、ノードBはノードAに対して、使用する接続のConnectIDと、開放されたチャンネルのChannelIDとをoperandフィールドに記述したレスポンスパケット30 ※

※を送る。具体的には、表11に示すように各フィールドが設定されたレスポンスパケット30が、ノードBからノードAへ返される。

【0111】

【表11】

| フィールド | cl | sub_l | c | ctype | operand[0] | operand[1] |
|-------|----|-------|---|----------|------------|------------|
| 設定値 | 1 | 0 | 0 | ACCEPTED | ConnectID | ChannelID |

【0112】コマンドパケット29及びレスポンスパケット30のやり取りにより、ノードAとノードBとの間で使用されていたチャンネルが開放される。なお、チャンネルの開放を行うコマンドパケット29及びレスポンスパケット30のやり取りは、チャンネルを開放しなくてもいい場合には、行う必要はない。

★る。このとき、ノードAはノードBに対して、開放しようとする接続のConnectIDをoperandフィールドに記述したコマンドパケット31を送る。具体的には、表12に示すように各フィールドが設定されたコマンドパケット31を、ノードAからノードBへ送出する。

【0114】

【表12】

【0113】次に、ノードAはノードBに対して、DISCONNECTコマンドを用いて、接続の開放を要求する★

| フィールド | cl | sub_l | c | ctype | opcode | operand[0] |
|-------|----|-------|---|---------|------------|------------|
| 設定値 | 1 | 0 | 0 | CONTROL | DISCONNECT | ConnectID |

【0115】コマンドパケット31により接続の開放がなされたら、次に、ノードBは、接続が開放されたことを示すレスポンスをノードAへ返す。このとき、ノードBはノードAに対して、開放された接続のConnectIDをoperandフィールドに記述した☆

☆レスポンスパケット32を送る。具体的には、表13に示すように各フィールドが設定されたレスポンスパケット32が、ノードBからノードAへ返される。

【0116】

【表13】

| フィールド | cl | sub_l | c | ctype | operand[0] |
|-------|----|-------|---|----------|------------|
| 設定値 | 1 | 0 | 0 | ACCEPTED | ConnectID |

【0117】コマンドパケット31及びレスポンスパ

ケット32のやり取りにより、ノードAとノードBとの間

で使用されていたコネクションが開放される。なお、コネクションの開放を行うコマンドパケット 3 1 及びレスポンスパケット 3 2 のやり取りは、コネクションを開放しなくてもいい場合には、行う必要はない。

【0 1 1 8】以上がノード A からのデータを複数のコマンドパケットに分割してノード B へ書き込む際の手順である。

【0 1 1 9】つぎに、上述したコマンドを用いて、ノード A によってノード B からデータを読み出す際の手順について、図 1 8 を参照して説明する。ここで、ノード B から読み出すデータは、複数のパケットに分割してノード B からノード A へ送信するものとしている。

【0 1 2 0】図 1 8 に示した手順では、まず、ノード A とノード B との間でコネクションを張り、次に、そのコネクションの中でチャンネルを確立する。次いで、読み出*

* し可能なデータサイズを確認した上で、複数のコマンドパケットを用いてノード B からデータの読み出しを行う。その後、チャンネルを開放し、更に、ノード A とノード B との間のコネクションを開放する。以下、この手順について、図 1 8 を参照して詳細に説明する。

【0 1 2 1】まず、ノード A がノード B に対して、CONNECT コマンドを用いて、コネクションを要求する。このとき、ノード A はノード B に対して、ノード A の ID 番号 NodeID(A) と、張ろうとするコネクションに対して割り当てた ConnectID とを operand フィールドに記述したコマンドパケット 4 1 を送る。具体的には、表 1 4 に示すように各フィールドが設定されたコマンドパケット 4 1 を、ノード A からノード B へ送出する。

【0 1 2 2】

【表 1 4】

| フィールド* | cl | sub_l | c | ctype | opcode | operand[0] | operand[1] |
|--------|----|-------|---|---------|---------|------------|------------|
| 設定値 | 1 | 0 | 0 | CONTROL | CONNECT | ConnectID | NodeID(A) |

【0 1 2 3】コマンドパケット 4 1 によりコネクションが正しく張れたら、次に、ノード B は、コネクションが正しく張れたことを示すレスポンスをノード A へ返す。このとき、ノード B はノード A に対して、ノード B の ID 番号 NodeID(B) と、割り当てられた ConnectID とを operand フィールドに記述したレスポンスパケット 4 2 を送 ※

※。具体的には、表 1 5 に示すように各フィールドが設定されたレスポンスパケット 4 2 が、ノード B からノード A へ返される。

【0 1 2 4】

【表 1 5】

| フィールド* | cl | sub_l | c | ctype | operand[0] | operand[1] |
|--------|----|-------|---|----------|------------|------------|
| 設定値 | 1 | 0 | 0 | ACCEPTED | ConnectID | NodeID(B) |

【0 1 2 5】以上のやり取りにより、ノード A とノード B との間でコネクションが張られる。なお、このやり取りは、ノード A とノード B との間で既にコネクションが張られている場合には、改めて行う必要はない。

【0 1 2 6】次に、ノード A はノード B に対して、OPEN コマンドを用いて、チャンネルの確立を求める。このとき、ノード A はノード B に対して、使用するコネクション ☆

☆ の ConnectID と、確立しようとするチャンネルに対して割り当てた ChannelID とを operand フィールドに記述したコマンドパケット 4 3 を送る。具体的には、表 1 6 に示すように各フィールドが設定されたコマンドパケット 4 3 を、ノード A からノード B へ送出する。

【0 1 2 7】

【表 1 6】

| フィールド* | cl | sub_l | c | ctype | opcode | operand[0] | operand[1] |
|--------|----|-------|---|---------|--------|------------|------------|
| 設定値 | 1 | 0 | 0 | CONTROL | OPEN | ConnectID | ChannelID |

【0 1 2 8】コマンドパケット 4 3 によりチャンネルが確立されたら、次に、ノード B は、チャンネルが確立されたことを示すレスポンスをノード A へ返す。このとき、ノード B はノード A に対して、使用するコネクションの ConnectID と、割り当てられた ChannelID とを operand フィールドに記述したレスポンスパケット 4 4 を送る。具体 ☆

☆ 的には、表 1 7 に示すように各フィールドが設定されたレスポンスパケット 4 4 が、ノード B からノード A へ返される。

【0 1 2 9】

【表 1 7】

| フィールド | cl | sub_l | c | ctype | operand[0] | operand[1] |
|-------|----|-------|---|----------|------------|------------|
| 設定値 | 1 | 0 | 0 | ACCEPTED | ConnectID | ChannelID |

【0130】以上のやり取りにより、ノードAとノードBとの間でチャンネルが確立される。なお、このやり取りは、ノードAとノードBとの間で既にチャンネルが確立されている場合には、改めて行う必要はない。

【0131】次に、ノードAはノードBに対して、READ 10 コマンドを用いて、ノードBから読み出しが可能なデータのデータサイズを問い合わせる。このとき、ノードAはノードBに対して、使用するコネクションのConnectI *

*Dと、使用するチャンネルのChannelIDとをoperandフィールドに記述したコマンドパケット45を送る。具体的には、表18に示すように各フィールドが設定されたコマンドパケット45を、ノードAからノードBへ送出する。

【0132】

【表18】

| フィールド | cl | sub_l | c | ctype | opcode | operand[0] | operand[1] |
|-------|----|-------|---|--------|--------|------------|------------|
| 設定値 | 1 | 0 | 0 | STATUS | READ | ConnectID | ChannelID |

【0133】次に、コマンドパケット45を受け取ったノードBは、ノードBから読み出しが可能なデータのデータサイズをノードAに知らせる。このとき、ノードBはノードAに対して、使用するコネクションのConnectIDと、使用するチャンネルのChannelIDと、ノードBから読み出しが可能なデータのデータサイズAvailableDataSiz※

※eとをoperandフィールドに記述したレスポンスパケット46を送る。具体的には、表19に示すように各フィールドが設定されたレスポンスパケット46を、ノードBからノードAへ送出する。

【0134】

【表19】

| フィールド | cl | sub_l | c | ctype | operand[0] | operand[1] | operand[2] |
|-------|----|-------|---|----------|------------|------------|-------------------|
| 設定値 | 1 | 0 | 0 | ACCEPTED | ConnectID | ChannelID | AvailableDataSize |

【0135】次に、ノードAはノードBに対して、READ コマンドを用いて、データの読み出しを要求する。このとき、ノードAはノードBに対して、使用するコネクションのConnectIDと、使用するチャンネルのChannelIDとをoperandフィールドに記述したコマンドパケット47を ☆

☆送る。具体的には、表20に示すように各フィールドが設定されたコマンドパケット47を、ノードAからノードBへ送出する。

【0136】

【表20】

| フィールド | cl | sub_l | c | ctype | opcode | operand[0] | operand[1] |
|-------|----|-------|---|---------|--------|------------|------------|
| 設定値 | 1 | 0 | 0 | CONTROL | READ | ConnectID | ChannelID |

【0137】次に、ノードBは、ノードAに要求されたデータをノードAへ送信する。ここで、ノードBから読み出されるデータは、n個に分割してノードAに送信されるものとし、ここでは、最初のデータであるReadingData1を送信する。このとき、ノードAはノードBに対して、使用するコネクションのConnectIDと、使用するチャンネルのChannelIDとをoperandフィールドに記述すると ☆

☆ともに、FCPdataフィールドの残りの領域にReadingData1を記述したレスポンスパケット48-1を送る。具体的には、表21に示すように各フィールドが設定されたレスポンスパケット48-1を、ノードAからノードBへ送出する。

【0138】

【表21】

| フィールド | cl | sub_l | c | ctype | opcode | operand[0] | operand[1] | |
|-------|----|-------|---|---------|--------|------------|------------|--------------|
| 設定値 | 1 | 0 | 1 | CONTROL | READ | ConnectID | ChannelID | ReadingData1 |

【0139】その後、同様に、ノードBからノードAへn個に分割されたデータの送信が順次行われる。このとき、sub_lフィールドの値は、データを分割して送信する各レスポンスパケットの順序を示すように、各コマンドパケット毎に1ずつ増加される。

【0140】そして、最後に、n個に分割されたデータのうちの最後のデータであるReadingDataNを送信する。このとき、ノードBはノードAに対して、使用する接続のConnectIDと、使用するチャネルのChannelIDとをoperandフィールドに記述するとともに、FCPdataフィールドの残りの領域にReadingDataNを記述したレスポ*

*ンスパケット48-nを送る。具体的には、表22に示すように各フィールドが設定されたレスポンスパケット48-nを、ノードBからノードAへ送出する。このとき、cフィールドの値は、コマンドパケット48-nの前のコマンドパケットまでは「1」とされるが、コマンドパケット48-nでは、このコマンドパケット48-nがn個に分割されたデータの送信を行うパケットうちの最後のパケットであることを示すように「0」とされる。

【0141】

【表22】

| フィールド | cl | sub_l | c | ctype | opcode | operand[0] | operand[1] | |
|-------|----|-------|---|---------|--------|------------|------------|--------------|
| 設定値 | 1 | n | 0 | CONTROL | READ | ConnectID | ChannelID | ReadingDataN |

【0142】そして、ノードBからのデータの読み出しが完了したら、次に、ノードAはノードBに対して、CLOSEコマンドを用いて、チャネルの開放を要求する。このとき、ノードAはノードBに対して、使用する接続のConnectIDと、開放しようとするチャネルのChannelIDとをoperandフィールドに記述したコマンドパケ ※

※ット49を送る。具体的には、表23に示すように各フィールドが設定されたコマンドパケット49を、ノードAからノードBへ送出する。

【0143】

【表23】

| フィールド | cl | sub_l | c | ctype | opcode | operand[0] | operand[1] |
|-------|----|-------|---|---------|--------|------------|------------|
| 設定値 | 1 | 0 | 0 | CONTROL | CLOSE | ConnectID | ChannelID |

【0144】コマンドパケット49によりチャネルの開放がなされたら、次に、ノードBは、チャネルが開放されたことを示すレスポンスをノードAへ返す。このとき、ノードBはノードAに対して、使用する接続のConnectIDと、開放されたチャネルのChannelIDとをoperandフィールドに記述したレスポンスパケット50 ★

★を送る。具体的には、表24に示すように各フィールドが設定されたレスポンスパケット50が、ノードBからノードAへ返される。

【0145】

【表24】

| フィールド | cl | sub_l | c | ctype | operand[0] | operand[1] |
|-------|----|-------|---|----------|------------|------------|
| 設定値 | 1 | 0 | 0 | ACCEPTED | ConnectID | ChannelID |

【0146】コマンドパケット49及びレスポンスパケット50のやり取りにより、ノードAとノードBとの間で使用されていたチャネルが開放される。なお、チャネルの開放を行うコマンドパケット49及びレスポンスパケット50のやり取りは、チャネルを開放しなくてもいい場合には、行う必要はない。

【0147】次に、ノードAはノードBに対して、DISC 50

ONNECTコマンドを用いて、接続の開放を要求する。このとき、ノードAはノードBに対して、開放しようとする接続のConnectIDをoperandフィールドに記述したコマンドパケット51を送る。具体的には、表25に示すように各フィールドが設定されたコマンドパケット51を、ノードAからノードBへ送出する。

【0148】

【表25】

| フィールド* | cl | sub_l | c | ctype | opcode | operand[0] |
|--------|----|-------|---|---------|------------|------------|
| 設定値 | 1 | 0 | 0 | CONTROL | DISCONNECT | ConnectID |

【0149】コマンドパケット51によりコネクションの開放がなされたら、次に、ノードBは、コネクションが開放されたことを示すレスポンスをノードAへ返す。このとき、ノードBはノードAに対して、開放されたコネクションのConnectIDをoperandフィールドに記述した*

* レスポンスパケット52を送る。具体的には、表26に示すように各フィールドが設定されたレスポンスパケット52が、ノードBからノードAへ返される。

【0150】

【表26】

| フィールド* | cl | sub_l | c | ctype | operand[0] |
|--------|----|-------|---|----------|------------|
| 設定値 | 1 | 0 | 0 | ACCEPTED | ConnectID |

【0151】コマンドパケット51及びレスポンスパケット52のやり取りにより、ノードAとノードBとの間で使用されていたコネクションが開放される。なお、コネクションの開放を行うコマンドパケット51及びレスポンスパケット52のやり取りは、コネクションを開放しなくてもいい場合には、行う必要はない。

【0152】以上がノードBからのデータを複数のコマンドパケットに分割して読み出す際の手順である。

【0153】なお、以上の説明では、IEEE1394バスを例に挙げて、本発明の実施の形態を説明してきたが、本発明はIEEE1394バス以外にも適用可能である。すなわち、本発明は、IEEE1394バスに限らず、従来は1つのパケットでコマンドを送ることを想定していたようなパケットで、コマンドの送受信だけでなく、1パケットに納まりきらないようなデータの送受信も行うようにしたいときに非常に有効である。

【0154】

【発明の効果】以上の説明から明らかなように、本発明によれば、コマンドの送受信だけでなく、1パケットに納まりきらないようなデータの送受信も、コマンドの送受信に使用されるものと同様なパケットによって効率良く行うことが可能となる。

【0155】特に、本発明をIEEE1394のAsynchronousパケットに適用したときには、Asynchronousパケットを用いても、1パケットに納まりきらないような大量のデータを、複数のパケットに分割して伝送することが可能となる。

【図面の簡単な説明】

【図1】本発明を適用して各機器を接続した状態を示すブロック図である。

【図2】1つのコマンドパケットに対して1つのレスポンスコマンドを返すときについて、コマンドパケットとレスポンスパケットのやり取りの手順を示す図である。

【図3】複数のコマンドパケットに対して1つのレスポ

ンスコマンドを返すときについて、コマンドパケットとレスポンスパケットのやり取りの手順を示す図である。

【図4】1つのコマンドパケットに対して複数のレスポンスコマンドを返すときについて、コマンドパケットとレスポンスパケットのやり取りの手順を示す図である。

【図5】書き込みデータを複数のコマンドパケットに分割して送り出すときについて、コマンドパケットとレスポンスパケットのやり取りの様子を示す図である。

【図6】読み出しデータを複数のコマンドパケットに分割して受け取るときについて、コマンドパケットとレスポンスパケットのやり取りの手順を示す図である。

【図7】Asynchronousパケットのフレーム構造を示す図である。

【図8】AV/Cコマンドフレームのフレーム構造を示す図である。

【図9】AV/Cレスポンスフレームのフレーム構造を示す図である。

【図10】本発明で使用するコマンドフレームの一例について、そのフレーム構造を示す図である。

【図11】本発明で使用するレスポンスフレームの一例について、そのフレーム構造を示す図である。

【図12】本発明で使用するコマンドフレームの他の例について、そのフレーム構造を示す図である。

【図13】本発明で使用するレスポンスフレームの他の例について、そのフレーム構造を示す図である。

【図14】図10に示したコマンドフレームを含むコマンドパケットの使い方の一例を示す図であり、3つのコマンドパケットに対して1つのレスポンスコマンドを返す場合を示す図である。

【図15】図11に示したレスポンスフレームを含むレスポンスパケットの使い方の一例を示す図であり、1つのコマンドパケットに対して3つのレスポンスコマンドを返す場合を示す図である。

【図16】本発明を適用したデータ伝送の一例につい

て、データ伝送の概念をモデル化して示す図である。

【図17】本発明を適用したデータ伝送手順の一例を示す図であり、データ書き込み時のコマンドパケットとレスポンスパケットのやり取りの手順を示す図である。

【図18】本発明を適用したデータ伝送手順の一例を示す図であり、データ読み出し時のコマンドパケットとレスポンスパケットのやり取りの手順を示す図である。

【図19】従来のAsynchronousモードにおけるパケットのやり取りを示す図であり、1つのコマンドパケットに対して1つのレスポンスコマンドを返す様子を示す図である。

【図20】従来のAsynchronousモードにおけるパケット*

*のやり取りを示す図であり、データ送信時のパケットのやり取りの手順を示す図である。

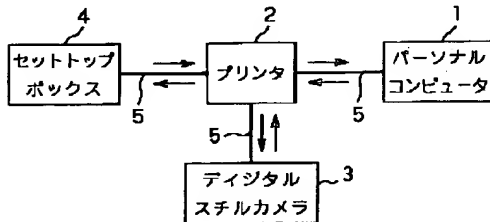
【図21】従来のAsynchronousモードにおけるパケットのやり取りを示す図であり、データ受信時のパケットのやり取りの手順を示す図である。

【図22】従来のAsynchronousモードで、送信データを複数のコマンドパケットに分割して送信するときのパケットのやり取りの手順を示す図である。

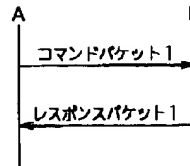
【符号の説明】

1 パーソナルコンピュータ、 2 プリンタ、 3 デジタルスチルカメラ、 4 セットトップボックス、 5 IEEE1394バス

【図1】

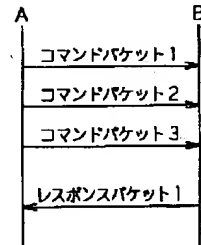


【図2】



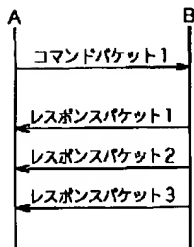
1コマンドパケット-1レスポンスパケット

【図3】



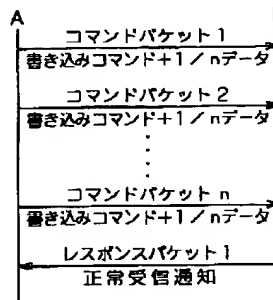
複数コマンドパケット-1レスポンスパケット

【図4】



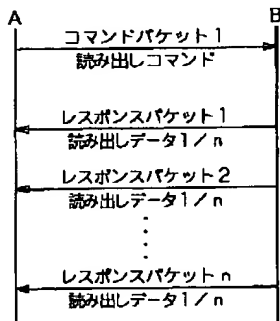
1コマンドパケット-複数レスポンスパケット

【図5】



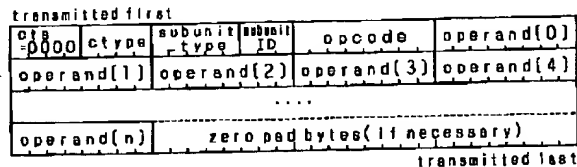
複数コマンドパケット-1レスポンスパケット(データ書き込み)

【図6】



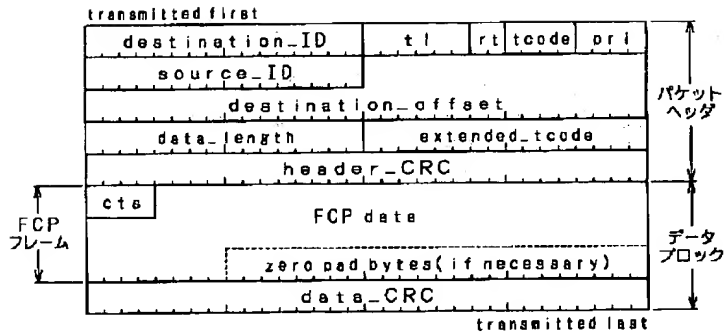
1コマンドパケット-複数レスポンスパケット(データ読み出し)

【図8】



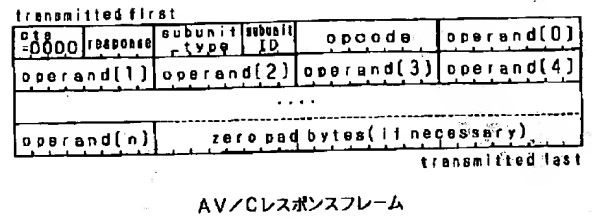
AV/Cコマンドフレーム

【図7】



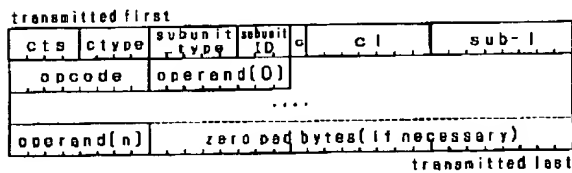
Asynchronousパケットのフレーム構造

【図9】



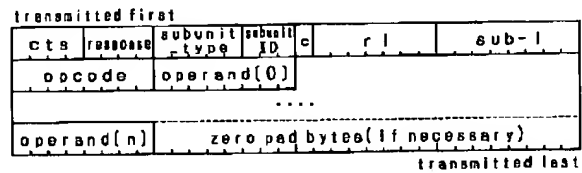
AV/Cレスポンスフレーム

【図10】



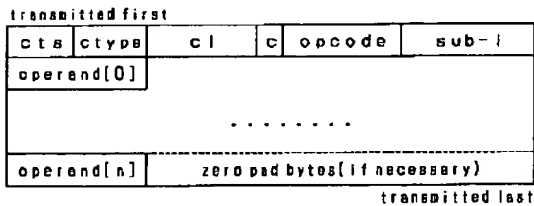
本発明で使用するコマンドフレームの一例

【図11】



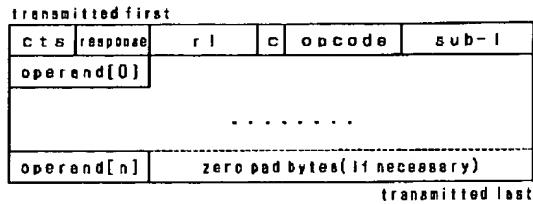
本発明で使用するレスポンスフレームの一例

【図12】



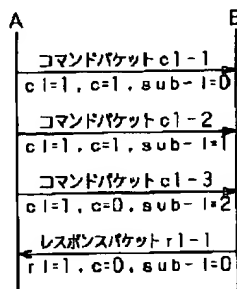
本発明で使用するコマンドフレームの他の例

【図13】

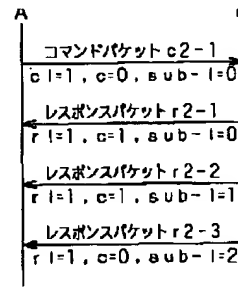


本発明で使用するレスポンスフレームの他の例

【図14】

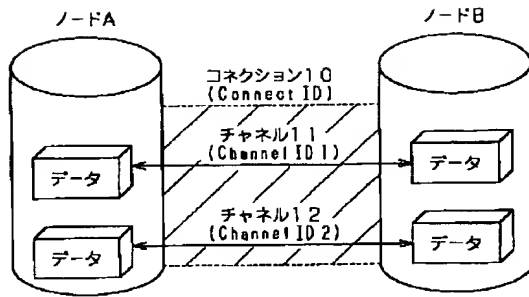


【図15】



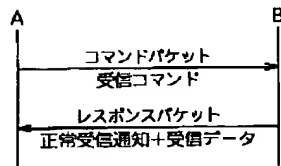
新規フィールドの使い方の一例(3コマンドパケット-1レスポンスパケット) 新規フィールドの使い方の一例(1コマンドパケット-3レスポンスパケット)

【図16】

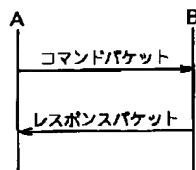


データ伝送の概念図

【図21】

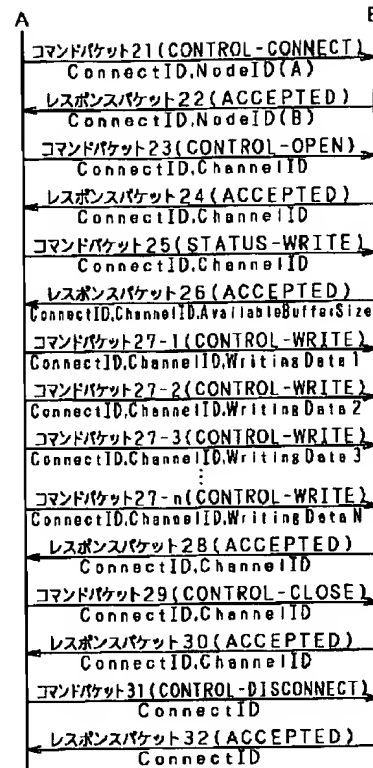
従来のAsynchronousモードにおけるパケットのやり取り手順
(データ受信の場合)

【図19】



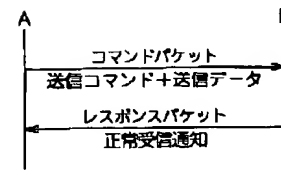
従来のAsynchronousモードにおけるパケットのやり取り手順

【図17】

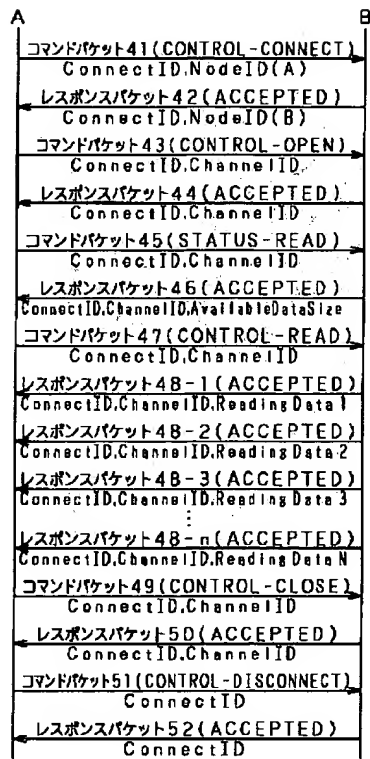


データ書き込みにおけるコマンドパケットとレスポンスパケットのやり取り手順

【図20】

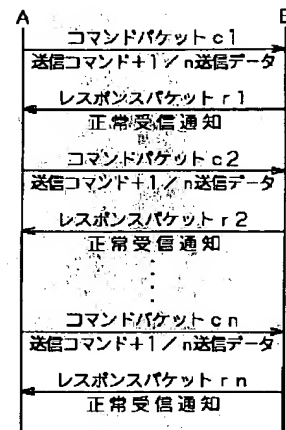
従来のAsynchronousモードにおけるパケットのやり取り手順
(データ送信の場合)

【図18】



データ読み出しにおけるコマンドパケットとレスポンスパケットのやり取り手順

【図22】



従来のAsynchronousモードにおけるパケットのやり取り手順
(n分割データ送信の場合)